CLIMA 2016 - proceedings of the 12th REHVA World Congress

volume 8

Heiselberg, Per Kvols

Publication date:
2016

Document Version
Publisher's PDF, also known as Version of record

Citation for published version (APA):
Heiselberg, P. K. (Ed.) (2016). *CLIMA 2016 - proceedings of the 12th REHVA World Congress: volume 8*. Department of Civil Engineering, Aalborg University.

Optimal Control of Complex HVAC Systems: Event-driven or Time-driven Optimization?

Junqi Wang[#1], Gongsheng Huang[#2], Yongjun Sun[*3]

[#]*Department of Architecture and Civil Engineering, City University of Hong Kong, Kowloon, Hong Kong*

[1]jqwang8-c@my.cityu.edu.hk

[2]gongsheng.huang@cityu.edu.hk

[*]*Division of Building Science and Technology, City University of Hong Kong, Kowloon, Hong Kong*

[3]yongjsun@cityu.edu.hk

Abstract

In large and complex HVAC systems, control optimization is always adopted to improve the operational efficiency since a small increase in the operational efficiency may lead to substantial energy savings. As the HVAC system becomes more and more complex, the real-time optimization of the system operation becomes a challenge due to the computational complexity. Almost all of the developed optimization methods are time-driven, in which the optimization is driven by "time" with a fixed optimization frequency. It is well-known that optimization should be done when the operational condition experiences a significant change, which may cause the current settings not optimal. Hence, "time" may not be the real optimization driver and the fixed optimization frequency may lead to unnecessary or delayed actions. To overcome these limitations, this paper proposes an event-driven optimization method, which originates from the event-driven control in control engineering. The key idea of the event-driven method is to use "event", rather than "time", as the optimization trigger. The "event" should be a well-defined condition, reflecting the system state or the state change. Optimizations will be conducted only when predefined events happened. The computation load and the energy saving of the proposed method are compared with that of a time-driven method by simulation. The results show that the computation loads of the proposed method are greatly reduced (up to 90%) compared with the time-driven method. The proposed method saves 10.65% of energy consumption based on the benchmark (no optimization is conducted), while the time-driven method saves 10.01%.

Keywords - HVAC optimal control; event-driven optimization; complex HVAC systems

1. Introduction

HVAC systems contribute the major part (20-50%) of building energy consumption [1]. It is worthwhile to consider the optimal control since a small increase in operating efficiency may lead to substantial energy savings [2], especially in complex HVAC systems. The optimal control of HVAC systems is achieved by finding optimal control settings and operation modes under dynamic working conditions [3]. With the increasing complexity of HVAC systems, the real-time control optimization of the system becomes a challenge in practice [4] because of the huge computation requirement. Plenty of optimization methods have been developed in the past either in a component level, a sub-systems level or a whole-system level. ASHRAE handbook [3] reviewed the publications since 1980s. New developments (till 2008) were surveyed by Wang and Ma [5].

Almost all of the developed methods belong to the type of time-driven optimization, in which the optimization is driven by "time" (often in a fixed frequency). For instance, in [6], the supply air temperature set-points and supply air static pressure set-points were updated every hour. A reduction of 7.66% in total energy consumption was achieved. Sun et al. [4] proposed a multiplexed optimization scheme which optimizes and updates one decision variable every 20 minutes. The results show that the computation load is drastically reduced together with the improved operation stability and energy performance.

These studies have demonstrated the potential savings associated with the control optimization in HVAC systems. However, it could be found that "time" may not be the real driver for a certain control optimization in HVAC systems. It is well-known that optimization should be done when the operational condition experiences a significant change, which may cause the current control settings or operation modes not optimal any more. Hence, defining the optimization driver based on the operational condition should be more appropriate than "time". Indeed, the fixed optimization frequency of the time-driven method may lead to unnecessary or delayed actions in real practices. For example, in the time-driven method, optimizations are also conducted when there is no need to (e.g., the operational condition is stable). This is surely an unnecessary wastage of resources like computation load and communication bus load [7]. Inevitably, this would also cause unnecessary changes of the actuator and therefore leads to unnecessary energy consumption as well as actuator attrition [8]. On the other hand, when there is a need to do optimizations (e.g., the operational condition experiences a critical change), optimizations may not be conducted on time because it did not happen exactly at the scheduled optimization time instants. Such delayed actions will cause the system operating at non-optimal settings for a period of time, which deteriorates the energy performance. Of course, the optimization frequency can be increased so that the delay can be reduced, but the computation load would also increase, which is bad. Meanwhile, frequent updating the set-points would cause stability issues and bring in disturbance to the control systems which, in turn, would also affect the system performance [9].

Considering the limitations mentioned above, the time-driven optimization may not be a very good solution. Thus, how to find a more suitable optimization driver, rather than simply using "time", becomes an interesting issue. With the recent development in control engineering, some new ideas are inspiring and some results are quite promising. As we known, the majority of the research and work in automatic control considers periodic control (equi-distant sampling interval), mainly due to the existence of a well-established system theory and sampled control systems (i.e., periodic or time-triggered) [10]. However, there are cases where it is interesting to consider event-driven control systems in which the action is event-triggered rather than time-triggered. Actually, what drives many of the processes are instantaneous “events” [11]. Much of the technology we have invented is event-driven, e.g., communication networks. In building energy control, actions are taken only when the networked sensors detect some “meaningful” changes in the environment, such as the temperature or humidity is passing some levels. Therefore, the HVAC optimal control problem actually has the event-driven nature. For applications, several case studies in control engineering have already shown that the computation load can be reduced effectively while still ensuring the control performance. For example, 50% of computation reduction was achieved in [12] and 70-80% of computation reduction was achieved in [13], while the control performance was guaranteed.

Unlike the conventional method that uses “time” to trigger the optimization action, this research aims to develop a control optimization method in which actions are triggered by the “event”. The “event” should be a well-defined condition, which can reflect the system state or the state change. The control policy is defined such that optimizations are conducted only when predefined events happened. The main contribution of this paper is that we demonstrate that the event-driven optimization works in a complex HVAC system, which suggests that the proposed method could be a more reasonable and efficient method for HVAC optimal control compared with the time-driven method. Section 2 presents the basic idea of the event-driven optimization together with the event definitions and the methodology of performance comparison. Case studies are given in Section 3, which contains the introduction of the HVAC system, the simulation platform, the mathematical problem formulation and the implementation procedure. Results and analyses are given in Section 4, while conclusions are drawn in Section 5.

2. Event-driven Optimization

2.1 Basic Idea

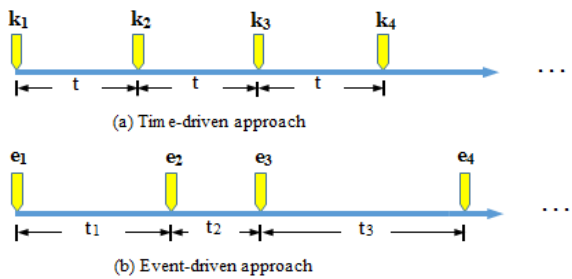

Fig. 1 Diagrams of time-driven and event-driven approaches. ($t1 \neq t2 \neq t3$)

The difference between the time-driven and the event driven optimization is illustrated in Fig. 1. The time-driven approach conducts optimizations in a fixed frequency (time interval “t” is equal) no matter how the condition changes, while the event-driven approach triggers optimizations only when certain events happened.

Normally, the time intervals between events are different. A possible advantage of the event-driven approach is that the optimization frequency could adapt to the changing operational condition (if events are properly defined), which may avoid unnecessary or delayed optimization actions that the time-driven approach may encounter.

2.2 Event Definitions

In the proposed event-driven approach, the event definition will drastically affect the optimization performance since it determines when to take actions. The overall strategy is defined as: when there is a “meaningful change” that will affect the system performance happened, an optimization is triggered; otherwise, no action is taken. Here, events are defined to capture the “meaningful change” in the HVAC system. In order to show the feasibility of the proposed method, two simple events are defined based on domain knowledge.

(1) Chiller on/off status change: The reason is that the load distribution will have a sudden change when turn on/off a chiller or chillers. Thus, an optimization is needed.

(2) Part-load ratio (PLR) change: The reason is that the PLR has a significant effect on the component (e.g., chiller [14]) efficiency. Basically, the algorithm will check the PLR every 20 minutes and compare the current PLR with the PLRs in the past 30 minutes. If the maximal deviation is greater than a threshold, an optimization is triggered. The initial threshold for PLR change is chosen to be 10%. Here, checking the PLR value every 20 minutes is used to prevent the potential frequent event occurrence.

2.3 Methodology of Performance Comparison

The performance of the proposed event-driven optimization method is evaluated by comparing with the conventional time-driven optimization method using computer simulation since it is cheaper and faster compared with experiments [15]. The virtual HVAC system representing the real building HVAC system (modeled in TRNSYS [16]) is used to produce the online operation data (i.e., status data), while the control settings are computed in a separate MATLAB platform. The computation load and the energy performance of different methods are compared based on two indices.

The first index is the time used to search the optimal solutions. The computation load is measured by the time required by the optimization method in searching the optimal solution. The smaller the time is, the lighter the computation load is. The functions *Tic* and *Toc* in MATLAB are used to record the time duration, where *Tic* recorded the start time of the optimization and *Toc* recorded the elapsed time. The second index is the daily energy saving percentage. This index is used to indicate the energy performance. A larger energy saving percentage means a better performance.

3. Case Studies

3.1 System Structure and Fundamental Controls

In case studies, the HVAC system consists of a condenser water loop, a chiller plant, a chilled water primary loop, a chilled water secondary loop and air distribution sub-systems. Fundamental controls of different sub-systems used in this study are presented as follows.

Chiller sequencing control determines which and how many chillers should be switched on or off according to the current load condition. Total cooling load based sequencing control is used, in which the sequence is determined according to the instantaneous cooling load Q_{ch} measured by (1). The calculated cooling load Q_{ch} is compared with the thresholds ($Q_z^{on/off}$) to determine the operation of chiller sequence. Generally, a dead band is adopted in this control to avoid frequent switch actions when the load is at the boundary. The switch-on/off thresholds are calculated in (2) and (3),

$$Q_{ch} = C_p M_w \left(T_{chw,rtn} - T_{chw,\sup}\right) \quad (1)$$

$$Q_z^{on} = z \times Q_{rated} \times \left(1 + dead\_band\right) \quad (2)$$

$$Q_z^{off} = \left(z - 1\right) \times Q_{rated} \times \left(1 - dead\_band\right) \quad (3)$$

, where C_p is the specify heat of water; M_w is the water mass flow rate; $T_{chw,rtn}$ and $T_{chw,sup}$ are the chilled-water return and supply temperature; Q_z^{on} is the switch-on threshold; Q_z^{off} is the switch-off threshold; z is the number of chillers in operating; Q_{rated} is the rated cooling capacity of chiller (here, each chiller has the same rated cooling capacity); *dead_band* is the dead band which is a user-defined number between 0 and 1. If the measured cooling load is larger than a predefined threshold and this state lasts for a period longer than a time limit, a chiller and its interlocked pump(s) will be switched on. If the measured cooling load is smaller than a predefined threshold and this state lasts for a period longer than a time limit, a chiller and its interlocked pump(s) will be switched off.

Cooling tower sequencing control determines which and how many cooling towers should be switched on according to the amount of the heat required to be rejected. The number of cooling towers N_{ct} is determined simply by the number of operating chillers as shown in (4), where α depends on system configuration.

$$N_{ct} = \alpha N_{ch} \quad (4)$$

Controls of critical temperatures: cooling-water supply temperature is controlled by adjusting the frequency of the cooling tower fans; chilled-water supply temperature from chiller(s) is controlled by modulating the flow rate of refrigerant inside the refrigeration cycle; chilled-water supply temperature from heat exchanger(s) is controlled through adjusting the water pump speed; supply air temperature is controlled through adjusting the chilled water flow rate through the cooling coils in the AHUs.

3.2 Simulation Platform (TRNSYS Model)

The TRNSYS model was established based on a real building in Hong Kong. The central chiller plant contains six identical water cooled centrifugal chillers with the rated capacity of 7230 kW. Each chiller is interlocked with two constant speed water pumps (i.e., chilled water pump in primary side and cooling water pump). The rated flow rates of the chilled water pump and cooling water pump are 345 l/s and 410 l/s respectively. Heat exchangers are adopted to deliver the chilled water from lower to upper floors. The returned chilled water is distributed evenly to the operating chillers.

Eleven identical cross-flow cooling towers are used with the nominal water flow rate of 250 l/s. The validated models of centrifugal chillers, cooling towers and pumps developed in [17, 18] are adopted. The standard component Type 699 and Type 508a in TRNSYS are directly used to model the heat exchangers and the AHUs. A time delay model, i.e., Type 661, is used to mimic the transportation time delay of the chilled water and the cooling water. In order to track the optimal set-points, several PI/PID controllers are used in local control loops. The first PI controller (P= -0.95, I=35s) is used to maintain the cooling-water supply temperature at its set-point. The second PID controller (P= -0.9, I=10s, D=5s) is adopted to control the chilled-water supply temperature in the secondary loop. The third PI (P= -0.3, I=2s) controller is used to control the supply air temperature from the AHUs. These PI/PID parameters are tuned through the trial-and-error method and kept unchanged when running with different optimization methods.

3.3 Mathematical Problem Formulation

The optimal settings for local control loops are optimized such that the overall system power consumption is minimized. For all-electric cooling without thermal storage (which is this case), minimizing power requirement at each point in "time" or "event" is equivalent to minimizing the power consumption [3]. Thus, the event-driven optimization problem can be mathematically represented as follows,

$$P_{sys,tot,e} = P_{ch,tot,e} + P_{ct,tot,e} + P_{pump,tot,e} + P_{fan,tot,e} = f\left(T_{cw,e}, T_{chw,prm,e}, T_{chw,\sec,e}, T_{sa,e}, U\right) \quad (5)$$

$$\left(T^*_{cw,e}, T^*_{chw,prm,e}, T^*_{chw,\sec,e}, T^*_{sa,e}\right) = \underset{T_{cw,e}, T_{chw,prm,e}, T_{chw,\sec,e}, T_{sa,e}}{\arg\min} P_{sys,tot,e} \quad (6)$$

, subject to operational constraints and comfort constraints; where *e* is the event; *P* is the power and *T* is the temperature; subscripts *sys, tot, ct, pump* and *fan* represent system, total, cooling tower, pump, and fan; subscripts *cw, chw, prm, sec* and *sa* represent cooling water, chilled water, primary, secondary and supply air; for instance, $T_{cw,e}$ is the cooling-water supply temperature set-point at event *e*; T^*_e is the optimal temperature set-point at event *e*; *U* is the vector of uncontrolled variables.

The operational constraints under the summer condition are shown in (7)-(10). Besides, two additional constraints are adopted as shown in (11) and (12). For simplicity, the indoor thermal comfort is assumed to be satisfied.

$$28°C \le T_{cw,e} \le 35°C \quad (7)$$

$$5°C \le T_{chw,prm,e} \le 8°C \quad (8)$$

$$6.5°C \le T_{chw,\sec,e} \le 10°C \quad (9)$$

$$12°C \le T_{sa,e} \le 18°C \quad (10)$$

$$\left|T_{e(k+1)} - T_{ek}\right| \le \Delta T_{Thres} = 0.5°C \quad (11)$$

$$T_{chw,prm,e} + 0.8°C \le T_{chw,\sec,e} \quad (12)$$

Equation (11) is used to prevent the system instability issues caused by dramatic set-point change. If the change is greater than the threshold, it is deliberately set to ΔT_{Thres} . Equation (12) is to ensure a minimal temperature difference between the primary and secondary sides of the chilled water loop. Please note that the system total power requirement can be written as a function of four controlled variables and uncontrolled variables (equation (5)) based on the component performance models.

To solve this control optimization problem, a simple search tool, exhaustive search, is used. As shown in (11), the temperature set-point difference cannot be larger than 0.5 °C at each updating. To search the optimal settings (i.e., a combination of four decision variables), a step change of 0.1 °C is used here, which is based on a previous study that also adopted the exhaustive search method [19].

3.4 Implement Issues

The implementation steps of the proposed event-driven optimization were executed at each time step and are listed in Table 1. The policy is simply defined as (step 3 in Table 1): if any of the two events happened, optimization actions will be taken. Otherwise, no action was taken. The simulation time step is 30 seconds and the overall simulation period is 24 hours. Uncontrolled variables are building cooling load, ambient air wet-bulb and dry-bulb temperature, which are derived from typical daily load and weather profiles of a real building in Hong Kong.

Table 1 Implementation steps

Function	Steps	Details
Check load condition	Step 1	Calculate the current PLR
Change operation mode	Step 2	Change the operation modes according to section 3.1 when necessary.
Check event occurrence	Step 3	If the “Chiller on/off status change” or “PLR change” happened, go to step 4. Otherwise, no action was taken.
Apply constraints	Step 4	According to the equations (7)-(12), generate all possible set-point combinations with 0.1 °C step change.
Control optimization	Step 5	Find out the set-point combination with the minimal system power requirement by exhaustive search.

4. Results and Analysis

4.1 Energy Performance and Computation Load Comparison

The energy performances of the time-driven method and the event-driven method are compared with a benchmark case in which the set-points were fixed at T_{cw}=30°C, $T_{chw,prm}$=6°C, $T_{chw,sec}$=7.5°C, T_{sa}=15°C.

In the time-driven optimization method, different optimization frequencies were tested. Firstly, it can be seen from Table 2 that the energy consumption saving increases as the optimization time interval decrease, which agrees well with the perception that higher optimization frequency leads to higher energy saving [20]. Secondly, using "Chiller on/off and PLR change" as the optimization trigger can save 10.65%, which is even better than conducting optimizations every 15 minutes (10.01%). A possible reason is that the time-driven optimization may have the delay when react to condition changes, while the event-driven method can take actions instantly without delay. Besides, using "Chiller on/off" only can already save 9.15%, which suggests that the "Chiller on/off" is an important event in terms of the energy saving.

Table 2 Energy performances and computation loads of different strategies

Op. methods	**Power consumption (kWh)**	**Power consumption saving**	**Op. times**	**Computation time (s)**	**Computation saving**
No Op.	225129	0.00%	0	0	/
One Op. / 2hrs	207599	7.79%	12	55.81	82.2%
One Op. / 1.5hrs	205929	8.53%	16	74.67	76.2%
One Op. / 1hrs	204518	9.16%	24	107.6	65.6%
One Op. / 30mins	203034	9.81%	48	198.9	36.5%
One Op. / 15mins	202604	10.01%	96	313.2	0.0%
Ch. On/Off	204534	9.15%	7	32.02	89.8%
Ch. On/Off & PLR Change	201150	10.65%	15	70.33	77.5%

(*Note*: Ch. = Chiller; Op. = Optimization; hrs = hours; mins = minutes.)

In the load comparison, "one optimization per 15 minutes" is used as the benchmark (as shown in Table 2), based on which the computation saving can be calculated. In the time-driven methods, the computation load decreases as the optimization frequency decreases, which is normal. Using the event "Chiller on/off" can save 89.8% of computation, while 75.5% of computation can be reduced by using the event "Chiller on/off and PLR change". It is noticed that the event-driven method only uses few times of optimizations (i.e., 7 and 15 times compared with 96 times in the benchmark), which results in a significant computation reduction. The reason is that the event-driven method has the ability to avoid unnecessary optimizations (when events are properly defined). For example, when the system is running stably, no action will be taken and control settings are kept unchanged.

4.2 Discussions

In Fig. 2, total power requirements of different methods are plotted against the simulation time. It is noticed that the total system power requirements of three optimization methods are very similar at the morning and evening periods, while considerable power requirement reductions can be observed from 9:00 to 18:00 by adopting control optimizations. Comparing with the time-driven method, the proposed event-driven method seems like finding a way that can further reduce the power consumption. If both energy savings and computation savings are considered in the comparison, the event-driven optimization should be better than the time-driven optimization since it outperforms almost all the cases of the time-driven method (Table 2). A main reason is that the event-driven method can avoid unnecessary optimizations. Besides, it has a quicker response to the critical condition changes since optimization actions can be taken instantly when events happened, while the time-driven method may have delay.

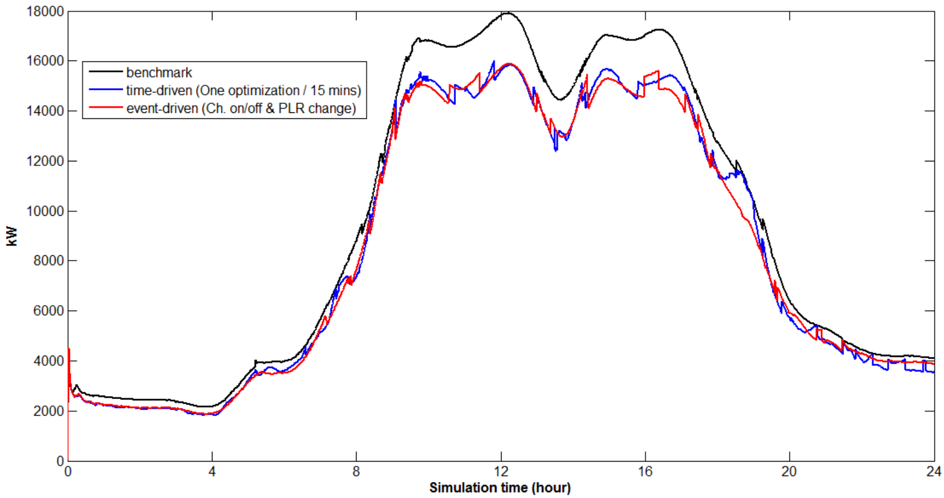

Fig. 2 System power requirements of different optimization methods

5. Conclusions

In traditional HVAC control optimization methods, “time” is used as the optimization driver. Intuitively, if we go one step deeper, we can find that “time” may not be the real optimization driver. In this research, an event-driven control optimization method has been proposed for complex HVAC systems with the aim to conduct optimizations in a more reasonable and efficient manner. The case studies suggest that events, like “Chiller on/off status change” and “PLR change”, can be used as the optimization triggers. The results show that the computation load of the proposed method can be greatly reduced (up to 90%) in comparison with the time-driven method. Compared with the benchmark case in which no optimization was used, the proposed method achieves a slightly higher daily energy saving percentage (10.65%) than the time-driven method (10.01%). It has been found that the control optimization can be performed in a more efficient manner with the proposed event-driven approach, and thus it is more practical for real-time applications. It should be noted that defining the proper “events” requires more sophisticated techniques compared to simply reacting to “time”. In the future, the event definitions will be further investigated and more events will be defined. For instance, different threshold values of the PLR change can be tested to analyze its effect on system performance.

Acknowledgment

The work described in this paper was fully supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. 11201215; CityU No. 9042177).

References

[1] L. Pérez-Lombard, J. Ortiz and C. Pout. A review on buildings energy consumption information. Energy Build., 40 (2008) 394-398.
[2] L. Lu, W. Cai, L. Xie, S. Li and Y. C. Soh, HVAC system optimization-in-building section, Energy Build., 37 (2005) 11-22,.
[3] ASHRAE. CHAPTER 42 SUPERVISORY CONTROL STRATEGIES AND OPTIMIZATION. ASHRAE Handbook - HVAC Applications., SI ed. Atlanta: USA: ASHRAE Inc., 2015.
[4] Y. Sun, G. Huang, Z. Li and S. Wang. Multiplexed optimization for complex air conditioning systems. Build. Environ., 65 (2013) 99-108.
[5] S. Wang and Z. Ma. Supervisory and Optimal Control of Building HVAC Systems: A Review. HVAC&R Research, 14 (2008) 3-32.
[6] A. Kusiak, M. Li and F. Tang. Modeling and optimization of HVAC energy consumption. Appl. Energy, 87 (2010) 3092-3102.
[7] J. Sandee, W. Heemels and P. Van Den Bosch. Case studies in event-driven control, Hybrid Systems: Computation and Control. Springer Berlin Heidelber, pp. 762-765, 2007.
[8] Q. Liu, Z. Wang, X. He and D. Zhou. A survey of event-based strategies on control and estimation. Systems Science & Control Engineering, 2 (2014) 90-97.
[9] D. Clarke and P. Gawthrop. Self-tuning control, Proc. of the Institution of Electrical Engineers, pp. 633-640, June 1979.
[10] K. Årzén. A simple event-based PID controller, Proc. 14th IFAC World Congress, 1999.
[11] C. G. Cassandras, Introduction to Discrete Event Systems. Springer Science & Business Media, 2008.
[12] J. H. Sandee. Event-driven control in theory and practice: Trade-offs in software and control performance (PhD thesis). Technische Universiteit Eindhoven, Netherlands, 2006.
[13] W. Heemels, J. Sandee and P. Van Den Bosch. Analysis of event-driven controllers for linear systems. Int J Control, 81 (2008) 571-590.
[14] S. B. Austin. Optimum chiller loading. ASHRAE J., 33 (1991) 40-43.
[15] Y. Pan, X. Lin, Z. Huang, J. Sun and O. Ahmed. A VERIFICATION TEST BED FOR BUILDING CONTROL STRATEGY COUPLING TRNSYS WITH A REAL CONTROLLER 2, Proc. of the Building Simulation: 12th Conf. of Int. Building Performance Simulation Association, Sydney, 2011.
[16] TRNSYS 16. A transient systems simulation program. http://sel.me.wisc.edu/trnsys/, 2006.
[17] S. Wang. Dynamic simulation of a building central chilling system and evaluation of EMCS on-line control strategies. Build. Environ., 33 (1998) 1-20.
[18] Z. Ma. Online supervisory and optimal control of complex building central chilling systems (PhD thesis). Hong Kong Polytechnic University, Hong Kong, 2008.
[19] Z. Ma and S. Wang. An optimal control strategy for complex building central chilled water systems for practical and real-time applications. Build. Environ., 44 (2009) 1188-1198.
[20] S. Huang and W. Zuo. OPTIMIZATION OF THE WATER-COOLED CHILLER PLANT SYSTEM OPERATION, 2014 ASHRAE/IBPSA-USA Building Simulation Conference, pp. 300-307, Atlanta, GA, 10-12 September 2014. ASHRAE.